

Serial No.: 10/632,084  
Response to Office Action Dated December 14, 2005  
Amendment Dated July 24, 2006

**BEST AVAILABLE COPY**

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (Currently Amended) A processor executing a plurality of instructions, comprising:  
  
an arithmetic logic unit; and  
  
a plurality of registers coupled to the ALU, each register programmable to store a register value;  
  
wherein said processor executes a test and skip instruction when called within the plurality of instructions that includes a first register reference and a second register reference and causes the processor to compare ~~the register a~~ first value comprising the register value stored in a register corresponding to the first register reference and a second value associated with the second register reference and to execute or not execute a subsequent instruction that follows the test and skip instruction based on the comparison.
2. (Original) The processor of claim 1 wherein the second value comprises a register value stored in the second register reference.
3. (Original) The processor of claim 1 wherein the processor is configured to access memory and the second value is stored in the memory.

**Serial No.: 10/632,084**  
**Response to Office Action Dated December 14, 2005**  
**Amendment Dated July 24, 2006**

4. (Original) The processor of claim 3 wherein the second register reference contains a value used to compute a pointer to a memory location containing the second value.
5. (Original) The processor of claim 4 wherein the pointer is computed by adding the value from the second register reference to a register value from another register.
6. (Currently Amended) The processor of claim 5 wherein the value held in the another register is post-incremented by a predetermined value following execution of the test and skip instruction.
7. (Original) The processor of claim 1 wherein the comparison includes a condition that is specified in the test and skip instruction.
8. (Original) The processor of claim 7 wherein any one of a plurality of conditions are specified in the test and skip instruction.
9. (Currently Amended) The processor of claim 7 wherein the condition is a condition selected from the group consisting of equal to, not equal to, less than, and greater than.
10. (Currently Amended) A method of executing a test and skip instruction, comprising:  
calling the test and skip instruction within a sequence of instructions;  
examining a bit in the test and skip instruction;

**Serial No.: 10/632,084**  
**Response to Office Action Dated December 14, 2005**  
**Amendment Dated July 24, 2006**

determining an address mode based on said bit;  
comparing contents of a first register to contents of a second register if the bit is in  
a first state; or  
comparing the contents of the first register to contents of a non-register location if  
the bit is in a second state; and  
skipping a subsequent instruction based on results of the comparison.

11. (Original) The method of claim 10 wherein skipping the subsequent instruction comprises replacing the subsequent instruction with a no operation instruction.

12. (Original) The method of claim 10 wherein the non-register location is a location selected from the group consisting of memory and a stack.

13. (Currently Amended) A system, comprising:

a main processor unit; and

a co-processor coupled to said main processor unit, wherein said co-processor executes a test and skip instruction when called within a sequence of instructions that includes a first register reference and a second register reference and causes the processor to compare the a first value comprising a register value stored in a register corresponding to the first register reference and a second value associated with the second register reference

**Serial No.: 10/632,084**  
**Response to Office Action Dated December 14, 2005**  
**Amendment Dated July 24, 2006**

and to execute or not execute a subsequent instruction that follows the test  
and skip instruction based on the comparison.

14. (Original) The system of claim 13 wherein the second value comprises a register value stored in the second register reference.

15. (Original) The system of claim 13 wherein the processor is configured to access memory and the second value is stored in the memory.

16. (Original) The system of claim 15 wherein the second register reference contains a value used to compute a pointer to a memory location containing the second value.

17. (Original) The system of claim 16 wherein the pointer is computed by adding the value from the second register reference to a register value from another register.

18. (Original) The system of claim 13 wherein the comparison includes a condition that is specified in the test and skip instruction.

19. (Original) The system of claim 18 wherein any one of a plurality of conditions are specified in the test and skip instruction.

**Serial No.: 10/632,084**  
**Response to Office Action Dated December 14, 2005**  
**Amendment Dated July 24, 2006**

20. (Currently Amended) The system of claim 18 wherein the condition is a condition selected from the group consisting of equal to, not equal to, less than, and greater than.

21. (Original) The system of claim 13 wherein the system comprises a communication device.

22. (Currently Amended) A programmable logic device comprising;

control logic; and

a means for executing a test and skip instruction when called within a sequence of instructions that includes a first register reference identifying a first register having a register value and a second register reference identifying a second register also having a register value, for comparing a first value comprising the register value stored in the first register and a second value associated with the second register, and for executing or not executing a subsequent instruction that follows the test and skip instruction based on the comparison.

23. (Currently Amended) The ~~system~~ programmable logic device of claim 22 wherein said second value is stored in a register.

24. (Currently Amended) The ~~system~~ programmable logic device of claim 22 wherein said second value is stored in memory.

**Serial No.: 10/632,084**  
**Response to Office Action Dated December 14, 2005**  
**Amendment Dated July 24, 2006**

25. (Currently Amended) The system-programmable logic device of claim 22 wherein said second value is stored in a stack.

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☐ BLACK BORDERS

☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES

☒ FADED TEXT OR DRAWING

☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING

☐ SKEWED/SLANTED IMAGES

☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS

☐ GRAY SCALE DOCUMENTS

☒ LINES OR MARKS ON ORIGINAL DOCUMENT

☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY

☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**